US Army Corps
of Engineers

The Hydrologic
Engineering Center

COED

Corps of Engineers Editor

February 1987

Draft

50272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO. DOD/SW/DK-89/003a	2.	3. Recipient's Accession No.

4. Title and Subtitle
COED, Corps of Engineers Editor

5. Report Date
Feb 87

6.

7. Author(s)
The Hydrologic Engineering Center

8. Performing Organization Rept. No.
CPD-56

9. Performing Organization Name and Address
Department of Army
The Hydrologic Engineering Center
Corps of Engineers
609 Second Street
Davis, California 95616

10. Project/Task/Work Unit No.
N/A

11. Contract(C) or Grant(G) No.
(C) N/A
(G)

12. Sponsoring Organization Name and Address
Department of Army
The Hydrologic Engineering Center, Corps of Engineers
609 Second Street
Davis, California 95616

13. Type of Report & Period Covered
Computer Program Documentation - Final

14.

15. Supplementary Notes
Copies available from HEC.
For diskette, see:

16. Abstract (Limit: 200 words)
COED is an editor containing full-screen editing capabilities, along with a powerful line-oriented command set that can be executed interactively or in a batch mode. COED is functional for most general text editing necessary for data preparation and program development. COED contains some additional features to aid in the preparation of data input files for several HEC programs. This includes right justifying data into predefined fields, and providing HEC program data variable names and definitions identified by the cursor location. The program is written in FORTRAN 77 with machine-dependent I-O. The line-edit mode is usable on any terminal. Full-screen mode is available for MS-DOS computers (or compatibles), and can be implemented on most 'smart' terminals.

Keywords: text editor, program documentation (KR)

17. Document Analysis a. Descriptors

b. Identifiers/Open-Ended Terms

c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Release Unlimited	Unclassified	70
	20. Security Class (This Page) Unclassified	22. Price

DO NOT PRINT THESE INSTRUCTIONS AS A PAGE IN A REPORT

INSTRUCTIONS

Optional Form 272, Report Documentation Page is based on *Guidelines for Format and Production of Scientific and Technical Reports, ANSI Z39.18–1974 available from American National Standards Institute, 1430 Broadway, New York, New York 10018.* Each separately bound report—for example, each volume in a multivolume set—shall have its unique Report Documentation Page.

1. Report Number. Each individually bound report shall carry a unique alphanumeric designation assigned by the performing organization or provided by the sponsoring organization in accordance with American National Standard ANSI Z39.23–1974, Technical Report Number (STRN). For registration of report code, contact NTIS Report Number Clearinghouse, Springfield, VA 22161. Use uppercase letters, Arabic numerals, slashes, and hyphens only, as in the following examples: FASEB/NS–75/87 and FAA/RD–75/09.
2. Leave blank.
3. Recipient's Accession Number. Reserved for use by each report recipient.
4. Title and Subtitle. Title should indicate clearly and briefly the subject coverage of the report, subordinate subtitle to the main title. When a report is prepared in more than one volume, repeat the primary title, add volume number and include subtitle for the specific volume.
5. Report Date. Each report shall carry a date indicating at least month and year. Indicate the basis on which it was selected (e.g., date of issue, date of approval, date of preparation, date published).
6. Sponsoring Agency Code. Leave blank.
7. Author(s). Give name(s) in conventional order (e.g., John R. Doe, or J. Robert Doe). List author's affiliation if it differs from the performing organization.
8. Performing Organization Report Number. Insert if performing organization wishes to assign this number.
9. Performing Organization Name and Mailing Address. Give name, street, city, state, and ZIP code. List no more than two levels of an organizational hierarchy. Display the name of the organization exactly as it should appear in Government indexes such as Government Reports Announcements & Index (GRA & I).
10. Project/Task/Work Unit Number. Use the project, task and work unit numbers under which the report was prepared.
11. Contract/Grant Number. Insert contract or grant number under which report was prepared.
12. Sponsoring Agency Name and Mailing Address. Include ZIP code. Cite main sponsors.
13. Type of Report and Period Covered. State interim, final, etc., and, if applicable, inclusive dates.
14. Performing Organization Code. Leave blank.
15. Supplementary Notes. Enter information not included elsewhere but useful, such as: Prepared in cooperation with . . . Translation of . . . Presented at conference of . . . To be published in . . . When a report is revised, include a statement whether the new report supersedes or supplements the older report.
16. Abstract. Include a brief (200 words or less) factual summary of the most significant information contained in the report. If the report contains a significant bibliography or literature survey, mention it here.
17. Document Analysis. (a). Descriptors. Select from the Thesaurus of Engineering and Scientific Terms the proper authorized terms that identify the major concept of the research and are sufficiently specific and precise to be used as index entries for cataloging.

    (b). Identifiers and Open-Ended Terms. Use identifiers for project names, code names, equipment designators, etc. Use open-ended terms written in descriptor form for those subjects for which no descriptor exists.

    (c). COSATI Field/Group. Field and Group assignments are to be taken from the 1964 COSATI Subject Category List. Since the majority of documents are multidisciplinary in nature, the primary Field/Group assignment(s) will be the specific discipline, area of human endeavor, or type of physical object. The application(s) will be cross-referenced with secondary Field/Group assignments that will follow the primary posting(s).
18. Distribution Statement. Denote public releasability, for example "Release unlimited", or limitation for reasons other than security. Cite any availability to the public, with address, order number and price, if known.
19. & 20. Security Classification. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED).
21. Number of pages. Insert the total number of pages, including introductory pages, but excluding distribution list, if any.
22. Price. Enter price in paper copy (PC) and/or microfiche (MF) if known.

☆ GPO : 1983 O - 381-526 (8393)

COED

Corps of Engineers Editor

February 1987

U. S. Army Corps of Engineers
Water Resources Support Center

The Hydrologic Engineering Center
609 Second Street
Davis, California 95616

FTS 460-1748 Commercial (916) 551-1748

Table of Contents

1.0 Introduction

COED is an editor containing full-screen editing capabilities, along with a powerful line-oriented command set that can be executed interactively or in a batch mode. COED is functional for most general text editing necessary for data preparation and program development. COED contains some additional features to aid in the preparation of data input files for several HEC programs. This includes right justifying data into predefined fields, and providing data variable names and definitions identified by the cursor location. For more information, see the Help Program (HP) command in section 3.6.

COED is available for a variety of computers and terminals. In the line-edit mode, it may be used on any terminal. In the full-screen mode, it is available for MS-DOS personal computers (or compatible), and can be implemented on any terminal conforming to the ANSI standards for terminals (a rate of 4800 baud or higher is recommended). (At the time of printing of this document, a full-screen version for some non-ANSI terminals was being planned.) COED is written in FORTRAN 77 and is transportable between machines. It has been implemented for Harris mini-computers, MS-DOS personal computers and compatibles.

1.0.1 New Features for this Version

A. The FILE, and SAVE commands can not be abbreviated; They each require all 4 characters to be specified.

B. LOCATE WILD (LW) and CHANGE WILD (CW) commands allow the use of 'wild' characters for locating and changing character strings.

C. Column commands allow a range of columns to be duplicated (CD), moved (CM), removed (CR), or set (CS) to a specific character.

D. CUT (CU) and SPLICE (SP) allow one line to be cut into two, or two to be spliced into one.

E. LOCATE EXCEPT (LE), and FIND EXCEPT (FE) commands will search for lines that do not contain a given character string.

F. X and Y commands have been expanded to include X1 through X5, and Y1 through Y5, adding 10 more X or Y commands than can be remembered.

G. On-line help (HE) has been added for the line edit and full screen edit modes.

H. The direction of movement through the file can be reversed by preceding certain commands by a minus '-' sign (e.g. -L /text/).

I. Type-ahead capability has been implemented for terminals on Harris Async ports.

J. A temporary backup copy of the file to be edited is automatically made.

1.1 Getting Started

The COED Editor is initiated by typing:

```
COED filename [cmdfile] [S=n]
```

where:

'filename' is the file that you wish to edit. If the file does not exist, COED will create the file.

'cmdfile' is an optional parameter providing the name of a file that contains a list of commands for COED to use. If this parameter is omitted commands are read from the keyboard, or standard input.

'S=n' is an optional parameter specifying the column length (n) of the lines to be edited. All lines are truncated to this length by COED. If this parameter is omitted, the first 100 lines of the file are examined to determine if a line length of 80 should be used. If characters are found beyond column 80, a column length of 133 will be used.

While COED is initializing, it makes a copy of your file for recovery purposes. If, for some reason, you save unwanted changes in your file, you can use this back copy to restore the file to its original condition. This must be done immediately after the edit session; if you log off, or edit another file, the backup copy will be destroyed. On Harris computers, this backup copy is kept in work file T1. On personal computers, the backup copy is named COED.BUP and is stored in the root directory (unless another directory is specified by using the "SETUP" command).

If COED is to be installed, refer to the appendix.

1.2 Recovery

In the event of a system crash during an edit session, the session can be recovered by using the recover option. This option uses two files to reconstruct the session up to the most recent command issued. One of the files is a copy of the original file before editing was begun. The other file is a record of all edit transactions performed by the editor.

To recover from a crash and resume editing, enter:

```
     COED.R                          (Harris systems only)

or   COED -R                         (All systems)
```

Note that no parameters, such as the filename or column length, may be used. When the recovery has been completed, the editor will be at the same position as at the time of termination. Zones and other attributes will all be the same as before the interruption.

WARNING: When continuing after a system halt on a Harris computer the user must log-on to the same port that was in use at the time of the halt (the same dialup port for dialup terminals). This is because the editor uses system work files that are identified by user and port number.

On MS-DOS systems, the recovery files are stored under the names COED.BUP and COED.REC in the root directory (unless another directory is specified by using the "SETUP" command). The COED.BUP file is a copy of the original file before the edit session was started. COED.REC is a record of all edit transactions made to the file during the edit session.

On Harris computers, the recovery files are stored in work files T1 and T2. Work file T1 is a copy of the original file before the edit session began. Work file T2 is a record of all edit transactions made to the file during the edit session.

1.3 Basic COED Commands

Command	Purpose	Examples
COED	Begin edit session	COED MYFILE COED OUTFILE S=132
FS	Go into Full Screen mode	FS
T	Go to Top of file (line zero)	T
B	Go to Bottom of file	B
P	Print line(s)	P (print 1) P 3 (print 3) P A (print all) P * (print to bottom)
N	Go to Next line	N (next line) N 3 (go down 3 lines)
U	Go Up	U 2 (go up 2)
L	Locate a string	L /ABC/
C	Change a string	C /ABC/XYZ/
I	Insert a line	I THIS IS A LINE
I	Input mode	I LINE ONE OF TWO LINE TWO OF TWO
$$$	Exit input mode	$$$
R	Replace a line	R NEW INFORMATION
DE	Delete a line	DE
CL	Show column banner	CL
FILE	Update file and exit	FILE NEWFILE
QU	Quit edit session (nothing saved)	QU

Note: A blank line (just a carriage return) causes the previous command to be repeated.

1.4 Command Syntax

This section describes the format and syntax of COED commands and how they are represented in this documentation.

COED prompts the user with 'E>' when it is ready for a command. The general form of a command is:

```
E>command [parameters]
```

The 'command' is always a command name followed by a blank or a comma, and any parameters, then a carriage return. Commands may be abbreviated to a 1, 2 or 4 letter command identifier. See the list of commands for valid command identifiers.

The syntax convention used in this document is represented in the following table.

Syntax Conventions

Symbol	Meaning
[]	Brackets surround optional parts of the commands.
|	Separates mutually exclusive parameter choices.
...	Three dots mean that parameters may be repeated.
CAP	Capital letters indicate that a specific parameter string must be used as shown.
/	Slashes are used as string delimiters in this documentation. However, any character may be used as a string delimiter so long as the character does not appear within the string.
c	Column number should be substituted.
m, n, r	Specifies a numeric value should be substituted.
*	The asterisk represents "all occurrences" in a numeric field and can be used in most places where an infinitely large numeric value is appropriate.
string (or stg)	Specifies a string of characters.
char	Specifies one character should be provided.

An example of the CUT command can illustrate the command syntax.

```
CU c|/string/ [n]
```

The command identifier is CU. It is shown in capital letters indicating it must be used exactly as given. A blank separates the command identifier from the parameter. The parameter, which is required, consists of two choices separated from each other by a vertical bar (|) symbol. This indicates the choices are mutually exclusive, that is one or the other must be used, but not both. The lowercase c indicates a numeric column number may be entered. The other possibility for this parameter is a string. If the string is used it must be delimited by a character such as the slash (/). The next parameter is optional because it is shown in brackets []. The n represents the number of lines upon which this command should operate. If the parameter n is not given, the default of one line is used.

```
CU 40 12
```

and

```
CU /THEN/
```

are both valid CUT commands.

1.5 Command Modifiers

Certain line-edit COED commands may be modified by use of special symbols:

Modifier	Purpose
.	Preceding a command by a period '.' suppresses the echo of the command results. This effectively turns off the verify option for that command only. See the verify command for more details.
=	Preceding certain commands by an equal sign '=' displays the last use of the command. Several COED commands have a "memory"; that is, they will remember the parameters from their most recent use. Commands that have a memory are: CHANGE, FIND, LOCATE, ZONE, X and Y. For example, if the same string is to be searched for several times with a LOCATE command, it can be specified the first time the LOCATE command is used; thereafter the L can be used to repeat the last LOCATE command. A equal sign used alone on a blank line will return current COED status information.
-	Preceding a command by a minus (-) reverses the direction of movement through the file for that command execution. The command "-L /ABC/" will locate the string "ABC" upwards, from the current line (until the top of the file is reached). Commands for which the minus sign can be used are CHANGE, LOCATE, FIND, and PRINT.
>	The greater than symbol (>) may be used in two ways: 1) If the last character of an INSERT, OVERLAY or REPLACE command is a '>' (followed by a carriage return) it indicates that the next line is to be placed at the end of the current line (a continuation line). The continuation character can be changed with the TERMDEF command. 2) If the '>' character is used within a line in an INSERT, OVERLAY, or REPLACE command the column location will move to the next tab stop (only the '>' is echoed).
<CR>	If a carriage return is entered (i.e., a blank line) the previous command issued will be repeated. This is very useful for repeating the Next, Locate, Print, DElete, and other commands.

1.6 Sample Commands

The following table provides examples of using the line-edit command syntax and modifiers.

Command	Description
HELP	Request help.
T	Move to top of file.
P 23	Print next 23 lines, starting at the current line.
C /ABC/XYZ/	Change ABC to XYZ on current line.
F C	Find next line beginning with "C".
.C /Z/AT/ * *	Change all occurrences of 'Z' to 'AT' from current position to end of file. Suppress the echo of changes.
-L /END/	Locate first occurrence of 'END' upward in the file.
FILE MYDATA	Store edit results in a file named 'MYDATA', and exit COED.
CU /THEN/	Cut the current line into two lines after the string 'THEN'.
SP 40 4	Splice (combine) the current line with the next line at column 40 (of the current line), and repeat this operation 4 times.
X ;L /ABC/;N;C /1/2/;	Set an X command to locate ABC, move to next line and change the first 1 to 2.
X 20	Execute the previously set X command 20 times.

2.0 Full Screen Edit Mode

2.1 Introduction

The COED full-screen edit mode enables text files to be edited by displaying a section of the text (about 20 lines worth) on the screen, and allowing the user to make changes by typing new or replacement text directly where desired. The four cursor keys and the page up, page down, home and end keys provide a means of movement around the file. This mode, combined with access to all of the line edit commands, provides a powerful text editing capability.

The full-screen mode operates in a "type-over" atmosphere: in the normal mode, whatever is typed replaces what is on the screen. The "Insert Character mode" allow characters to be inserted without erasing other characters. The "Insert Line mode" provides a means of inserting lines in the text file. In a new file, or at the bottom of an existing file, a carriage return provides an implicit new line. Thus to enter text into a new file in the full-screen mode, all one has to do is to type normally. Any of the line edit commands can be accessed from the full-screen mode by pressing the line-edit function key.

The full-screen feature is currently available for terminals that meet ANSI standards, and for MS-DOS compatible personal computers. Work is currently under way to provide full screen editing for some "smart" terminals that do not meet the ANSI standards.

2.2 Initiating the Full Screen Mode

When COED is executed, it enters the line-edit mode. To go into the full screen mode, the FULL SCREEN command must be entered. The syntax of the FULL SCREEN command is:

```
E>FS [L][C]
```

On Harris computers, with a terminal that meets ANSI standards, the "L" parameter will cause the current line number to be displayed on the status line. The "C" parameter will cause the current column number to be displayed on the status line. Due to transmission (baud) rates, these parameters can cause slower cursor movement on some terminals. Either (or both) parameters can be set permanently in the COED terminal definition file (SYST*COEDTD).

The Keypad keys (located on the right side of the keyboard) are used for moving around the file, and inserting and deleting characters. Function keys (located on the left side of the most personal computer keyboards, and along the top of many terminal keyboards) are used for editing functions such as deleting lines, inserting lines and issuing line-edit commands. On ANSI terminals, the function keys (if available) are located in different areas. Their functions are defined (and can be changed) in the COED terminal definition file (SYST*COEDTD).

2.3 Full Screen Key Descriptions.

2.3.1 Movement Keys.

Key	Description
CURSOR RIGHT	Moves the cursor right one character
CURSOR LEFT	Moves the cursor left one character
CURSOR UP	Moves the cursor up one line. At the top of the screen, the screen is scrolled down one line.
CURSOR DOWN	Moves the cursor down one line. At the bottom of the screen, the screen is scrolled up one line.
PAGE UP	Moves up approximately one screen
PAGE DOWN	Moves down approximately one screen.
HOME	Pressed once, goes to the top left corner of the screen. Pressed again, goes to the first line in the file.
END	Pressed once, goes to the bottom left corner of the screen. Pressed again, goes to the last line in the file.
BEGINNING OF LINE	Pressing the Keypad keys 5 then 4 (cursor left), moves the cursor to column one of the current line.
END OF LINE	Pressing the Keypad keys 5 then 6 (cursor right), moves the cursor to one column past the last character on the current line.
TOP OF SCREEN	Pressing Keypad keys 5 then 8, moves the cursor to the top of the screen in the current column.
BOTTOM OF SCREEN	Pressing Keypad keys 5 then 2, moves the cursor to the bottom of the screen in the current column.
SCROLL UP	Causes the screen to scroll up one line.
SCROLL DOWN	Causes the screen to scroll down one line.

2.3.1 Movement Keys (continued). (Some of the following keys are in different locations for different terminals.)

Key	Description
NEXT WORD	Moves the cursor to the beginning of the next word on the right. Words are delimited by spaces, commas, quote marks, parenthesis, etc.
PREVIOUS WORD	Moves the cursor to the beginning of the previous word on the left. Words are delimited by spaces, commas, quote marks, parenthesis, etc.
TAB	Moves the cursor to the tabs settings used in line-edit mode.
TAB LEFT	Moves the cursor one tab setting to the left.
BACKSPACE	Moves back one position and blanks that space.
CARRIAGE RETURN	Puts the cursor on the next line at the left margin. A new line will be generated if either in insert line mode, or if at the bottom of the file.

2.3.2 Editing Keys.

Key	Description
INSERT CHARACTER MODE	Toggles you into or out of insert character mode. In insert character mode, whenever characters are pressed, they are inserted to the left of the cursor. A backspace will delete the character to the left of the cursor in this mode. New lines are not created by insert character.
DELETE CHARACTER	Deletes characters on the current line. Delete character may be used in two ways: a. Delete highlighted area: Press delete character key, move cursor to highlight string to be deleted, press delete character key again to delete. A string may be highlighted by pressing the return key, End of line key, Beginning of line key, or a regular character (whereby the cursor will index right to that character. b. Delete single characters: Single characters are deleted when the delete key is pressed a second or subsequent time.
DELETE LINE	Deletes the line the cursor is on.
INSERT LINE MODE	Toggles into and out of insert line mode. Pressing the insert line key will provide you with a new line. Additional lines are generated by a carriage return, until insert line mode is turned off. If you are at the bottom of the file, a carriage return alone creates a new line (it will not put you into insert line mode, however.)
RESTORE	Restores the line you are on, if the previous command was in error. If the last key pressed was 'Delete line', that line will be resorted. If you inadvertently mistyped characters on the line, are in the process of deleting character (or have deleted characters), restore will return the line to what it was before you positioned the cursor on it. Once you have moved off of the line, it cannot be restored.
COMMAND	Provides an E> to execute one command, then returns to full screen mode. This is useful for doing locates, etc.
LINE-EDIT	Returns the program to the regular line-edit mode. To return to full screen mode, enter 'FS' again.

2.3.3 Edit Control Keys.

Key	Description
HELP	The HELP key provides access to on-line documentation of COED, including the locations of full screen edit keys. To return to editing, press the HELP key again.
NUM-LOCK	When NUM-LOCK is on, the Keypad enters into the numeric Keypad mode, disabling the cursor controls on the Keypad. "NUM-LOCK" will appear on the status line when this is on.
FILE	The FILE key will display a prompt with the current file name. To store the edited file under this name, press a carriage return. To store the file under a different name, type the new name. After storing the file, COED terminates.
SAVE	The SAVE key will display a prompt with the current file name. To store the edited file under this name, press a carriage return. To store the file under a different name, type the new name. After storing the file, COED returns to the edit session.
QUIT	The QUIT key will display a prompt that asks for a carriage return to quit. This will terminate COED without saving any changes to the file. To not QUIT (return to the edit session), press escape, or type "NO".
ERASE FIELD	The ERASE FIELD key erases all the characters in the current field (where the cursor is located). The field is defined by the tab settings.
HELP VARIABLE	When the Help Program feature has been enabled, the HELP VARIABLE key displays the definition of the variable for the current field.
JUSTIFY	The JUSTIFY key toggles the justification setting between either right (data) justification and no justification. This key is generally used only when preparing a data input file without the Help Program command on (see the Help Program appendix).

Key	Description
REFRESH SCREEN	The REFRESH SCREEN key repaints the screen. This key may be used when the screen is accidentally cleared or changed locally (at the terminal).
SETUP (PC only)	The SETUP key on the personal computer provides a means of setting parameters that will be saved across editing sessions. This includes screen colors, the directory of accessory files (e.g. the help file and recovery files), and an option causing COED to enter the full-screen mode automatically after initiated. Follow the directions printed on the screen, after pressing this key.

2.4 Key Locations

2.4.1 IBM P.C. and Compatibles

A key location template for the function keys and keypad key is located in the appendix.

A. Keypad. The keypad is located on the right side of the keyboard.

Key	Function
.	Delete Character
0	Insert Character Mode
1	End
2	Cursor Down
3	Page Down
4	Cursor Left
6	Cursor Right
7	Home
8	Cursor Up
9	Page Up
-	Scroll Up
+	Scroll Down
5 then 2	Bottom of Screen
5 then 4	Beginning of Line
5 then 6	End of Line
5 then 8	Top of Screen
Ctrl 4	Previous Word
Ctrl 6	Next Word

B. Function Keys. The function keys are the 10 keys marked F1 through F10 either on the left side of the keyboard, or along the top of the keyboard.

Key	Function
F1	Help
Alt F1	Help Variable
F2	Restore
Ctl F2	PC Setup
F3	Delete Line
F4	Insert Line Mode
F5 - F8	Reserved
F9	Single Command
F10	Line Edit Mode
Alt F10	File
Sft F10	Save
Ctl F10	Quit

2.4.2 Terminals (Meeting ANSI Standards)

A. Keypad. The keypad is located on the right side of the keyboard. A key location template for the keypad is provided in the appendix.

Key	Function
.	Delete Character
0	Insert Character Mode
1	End
2	Cursor Down
3	Page Down
4	Cursor Left
6	Cursor Right
7	Home
8	Cursor Up
9	Page Up
-	Scroll Up
, (tab)	Scroll Down
5 then 2	Bottom of Screen
5 then 4	Beginning of Line
5 then 6	End of Line
5 then 8	Top of Screen
"enter"	Next Word
5 enter	Previous Word

B. Function Keys. The number and location of function keys vary among terminals (these are defined in the SYST*COEDTD file). Function key identifications may be shown by pressing the HELP key from the full screen mode. Function key templates for some ANSI terminals are provided in the appendix. See your site manager for more information. The following table provides control characters that will provide the same functions.

Control Character	Function
^A	Single Command
^B	Line Edit Mode
^C	Restore Line
^D	Help
^F	Insert Line Mode
^G	Delete Line
^N	Num-Lock

3.0 Command Documentation

3.1 Basic Line Movement Commands

3.1.0 Summary

Name	Use
Top	T
Bottom	B
Next	N [n]
Up	U [n|/string/]
Print	P [n|ALL]
Locate	L [/string/] [AND|OR|NOT /string2/] [n]
Find	F [string]
Goto	G n

3.1.1 TOP

T

The TOP command moves the line pointer to the top of the file. This line is defined as line number zero and is not saved with the file (this line allows for insertion of new lines at the top of the file). Line zero cannot be changed, deleted, etc. The top of file identifier "TOF.." is displayed here.

3.1.2 BOTTOM

B

The BOTTOM command moves the line pointer to the bottom of the file. The end of file identifier "EOF.." is displayed here.

3.1.3 NEXT

N [n]

The NEXT command moves the line pointer down n lines. The default number of lines is 1.

3.1.4 UP

U [n|/string/]

The UP command moves the line pointer up n lines. If n is not specified, 1 is assumed. If a string is specified, the UP command searches upward for "string", starting with the line above the current line.

3.1.5 PRINT

P [n|ALL]

The PRINT command prints the next n lines, starting with the current line. If "*" is used for the line count n, the remainder of the file, beginning with the current line, is printed. In these cases, the line pointer is moved to the last line printed. If the "ALL" option is used, the entire file is printed, but the line pointer is not affected. "ALL" may be abbreviated to "A". To stop execution of the PRINT command, press control-X. Prefixing the PRINT command with a minus sign (-P) will print upward n lines.

3.1.6 LOCATE

L [/string/] [AND|OR|NOT /string2/] [n]

The LOCATE command searches for the next n occurrences of "string" beginning with the next line. If n is not specified, only the next occurrence of "string" is located. If an asterisk (*) is given for n, all occurrences of "string" will be located (beginning with the next line). The pointer is positioned to the line containing the string. If the string is not found, the pointer is positioned to the last line in the file. The LOCATE command without any parameter will repeat the most recent LOCATE command. Prefixing the LOCATE command with a minus sign "-L" will locate the specified string upward from the current line.

LOCATE can also be used with the operators "AND", "OR", and "NOT" along with a second string to find lines which meet one of these three conditions. The syntax is:

L /str1/OR/str2/ (Locate the next line containing the string "str1" or the string "str2".)

L /str1/AND/str2/ (Locate the next line containing both string "str1" and string "str2".)

L /str1/NOT/str2/ (Locate the next line containing the string "str1", but not containing the string "str2".)

As an example, if one wants to determine what FORTRAN subroutines the variable "XYZ" is located in, the following LOCATE command can be given:

L /SUBROUTINE/OR/XYZ/ *

3.1.7 FIND

F [string]

The FIND command searches each line, beginning with the next line, attempting to match each character of "string", with an identical string which starts in the first column of the line. When found, the line pointer is repositioned to that line. The first character of string is the next character following the single blank after the command name. Only non-blank characters in string are compared. The FIND command without any parameter will repeat the most recent FIND command. String delimiters are not used.

3.1.8 GOTO

G n

The GOTO command moves the line pointer to line number n, where n is a decimal number with respect to the top of the file.

3.2 Basic Line Edit Commands

3.2.0 Summary

Name	Use
Change	C [/stg1/stg2/ [n|n m]]
DElete	DE [n]
Insert	I [line]
Replace	R line
Overlay	O line

3.2.1 CHANGE

C [/stg1/stg2/ [n|n m]]

The CHANGE command replaces the m-th occurrence of "stg1" with "stg2" for the next n lines. The default value for m and n is 1. A "*" used for n denotes all lines from the current line to the bottom of file; a "*" used for m denotes all occurrences of "stg1". The line pointer will advance n-1 lines toward the bottom of the file. The CHANGE command without any parameter will repeat the most recent CHANGE command.

Prefixing the CHANGE command with a minus sign (-C) will reverse the pointer direction and will replace the mth occurrence of "stg1" with "stg2" up through the file for n lines.

3.2.2 DELETE

DE [n]

The DELETE command deletes n lines from the file starting with the current line. If n is omitted, just the current line will be deleted. An asterisk (*) used for n will delete all lines from the current line to the bottom of file.

3.2.3 INSERT

I [line]

The INSERT command inserts "line" after the line at which the pointer is currently positioned, and advances the pointer to the new line. If no string follows the INSERT command, the editor enters the "input" mode. In the input mode, the prompt is changed to "I>". All lines entered thereafter are inserted in the file directly below the current line. The input mode is terminated by typing three dollar signs ($$$) at the beginning of the line. Upon exit from the input mode, the line pointer is positioned to the last line inserted.

3.2.4 REPLACE

R line

The REPLACE command replaces the current line with "line".

3.2.5 OVERLAY

O line

The OVERLAY command places the non-blank characters of "line" in the corresponding positions of the current line. The first character of line is the next character following the single blank after the command name.

3.3 Terminate Edit Commands

3.3.0 Summary

Name	Use
FILE	FILE [filename]
SAVE	SAVE [filename]
QUIT	QU

3.3.1 FILE

FILE [filename]

The FILE command stores the edited file on the disk, replacing the existing copy of the file. Following the update of the file, COED exits. If an optional file name is specified, the edited file is stored in the specified disk file, leaving the original file unmodified.

3.3.2 SAVE

SAVE [filename]

The SAVE command stores the updated copy of the file on the disk, replacing the existing copy. If an optional file name is given, the edited file is stored in this file, leaving the original file unmodified. The current line pointer not affected, and the editor is not exited.

3.3.3 QUIT

QU

The QUIT command causes COED to exit without updating the file. The user should be sure to use QUIT only when the edit session results are to be discarded. The recovery option may be used when a QUIT is accidently requested.

3.4 Block Commands

3.4.0 Summary

Name	Use	
STart	ST	
ENd	EN	
DUplicate	DU	[n]
MOve	MO	
REmove	RE	
GEt	GE	[filename] [m|m n]
PUt	PU	[filename]

3.4.1 START

ST

The START command defines the current line as the beginning of a block.

3.4.2 END

EN

The END command defines the current line as the end of a block. A START command must have been entered to define the beginning of the block before an END is given.

3.4.3 DUPLICATE

DU [n]

The DUPLICATE command copies the previously defined block, inserting it directly below the current line. If the parameter n is given, the block is copied n times (n defaults to 1). The DUPLICATE command may be repeated without redefining the start and end of the block. The line pointer remains in the same position.

3.4.4 MOVE

MO

The MOVE command removes the previously defined block from its former location and inserts it directly below the current line. The line pointer remains in the same position.

3.4.5 REMOVE

RE

The REMOVE command eliminates the defined block from the file.

3.4.6 GET

GE [filename] [m|m n]

The GET command inserts line m or lines m through n of the specified file directly below the current line. If m and n are omitted, the entire file is inserted. If the file name is omitted the file previously specified will be used. The line pointer will be positioned at the inserted line or the last line of the inserted block.

3.4.7 PUT

PU [filename]

The PUT command copies the defined block to another file. If the file specified does not exist, it will be created. If the file already exists, the block is appended to the file. If the file name is omitted the file previously specified will be used.

3.5 Advanced Line Edit Commands

3.5.0 Summary

Name	Use	
ALter	AL	
CoLumn scale	CL	
Column Duplicate	CD	m-n c [r]
Column Move	CM	m-n c [r]
Column Remove	CR	m-n [r]
Column Set	CS	m[-n] "char" [r]
Locate Except	LE	/string/
Find Except	FE	string
Locate Wild	LW	/string/ [n]
Change Wild	CW	/string1/string2/ [n|n m]
Define Wild chars	DW	[c1 c2]
CUt line	CU	c|/string/ [n]
SPlice lines	SP	c|/string/ [n]
FRee	FR	[OFF|VERIFY]
TRuncate	TR	c [n]
X	X	[/command/[command/[...]]|n]
Y	Y	[/command/[command/[...]]|n]

3.5.1 ALTER

AL

The ALTER command allows one to change, delete, or insert characters correspondingly to their column position in a line. This is accomplished by printing the current line, then a new line with no prompt. The user then spaces out and types those characters to be changed, according to the following rules:

1) A space leaves the corresponding position unchanged.
2) A character replaces the character in the corresponding position.
3) A "#" replaces the corresponding character with a blank.
4) A "@" deletes the corresponding character and compresses the line.
5) A "%" inserts one blank before the corresponding character.
6) A "%/string/" inserts the string before the corresponding character.

After any changes, the line is printed again, and more changes can be made, or a carriage return with no changes will terminate the ALTER mode.

3.5.2 COLUMN SCALE

CL

The COLUMN SCALE command will print the current line and a column scale, beginning in column one, as an aid to identifying column position. If control characters appear in the line, it is displayed by a caret (^) with the control character under it.

3.5.3 COLUMN DUPLICATE

CD m-n c [r]

The COLUMN DUPLICATE command will duplicate the inclusive range of columns (m-n) to the position after column (c) for (r) lines. If (r) is omitted, the duplication will occur only on the current line. For example, "CD 40-70 0 4" will duplicate columns 40 through 70 to the position after column 0 for 4 lines.

3.5.4 COLUMN MOVE

CM m-n c [r]

The COLUMN MOVE command will move the inclusive range of columns (m-n) to the position after column (c) for (r) lines. If (r) is omitted, the move will occur only on the current line. For example, "CM 1-9 20 2" will move columns 1 through 9 to the position after column 20 for 2 lines.

3.5.5 COLUMN REMOVE

CR m-n [r]

The COLUMN REMOVE command will remove the inclusive range of columns m through n from (r) lines. If (r) is omitted, the remove will occur only on the current line. For example, "CR 20-29 3" will remove columns 20-29 (inclusively) for the current line and next two lines.

3.5.6 COLUMN SET

CS m[-n] "char" [r]

The COLUMN SET command will set the inclusive range of columns m through n (or just column m if -n is excluded), to the single specified character "char" for r lines. If (r) is omitted, the set will occur only on the current line. For example, "CS 73-80 ' ' *" will set columns 73-80 to blank on all lines, and "CS 73 H" will set column 73 to 'H' on the current line. Single quotes are required around a blank, a comma, a single quote, or other delimiter.

3.5.7 LOCATE EXCEPT

LE /string/ [n]

The LOCATE EXCEPT command operates similar to the LOCATE command, except that the lines where "string" does not appear are located. The LOCATE EXCEPT command searches for the next n times where "string" does not appear in the line, beginning with the next line. If n is not specified, only the next non-occurrence of "string" is located. If an asterisk (*) is given for n, all non-occurrences of "string" will be located (beginning with the next line). The pointer is positioned to the last line with a non-occurrence The memory will be retained until a LOCATE (or another LOCATE EXCEPT) command is given.

3.5.8 FIND EXCEPT

FE string

The FIND EXCEPT command operates similar to the FIND command, except that this command finds strings that do not match. The FIND EXCEPT command searches each line, beginning with the following line, attempting to find a non-match for each character in the string. When this non-match occurs, the line pointer is repositioned to that line. The first character of the string is the next character following the single blank after the command name. Only non-blank characters in the string are compared. String delimiters are not used. The find memory will be retained until a FIND (or another FIND EXCEPT) command is issued.

3.5.9 LOCATE WILD

LW /string/ [n]

The LOCATE WILD command allows one to locate string using wild characters that will match other characters. Two wild characters are available, a "?" and a "*". A "?" in the string will match any other single character. A "*" in the string will match any number of characters (including zero). For example:

LW / IF*THEN / will locate the following strings:

```
IF (X.GT.Y) THEN
IF (LFIRST) THEN
IFTHEN
```

LW /XARY(?)=/ will locate the following strings:

```
XARY(1) = 2 * X * 3.1416
XARY(4) = Y / XARY(3)
```

There may be up to 40 wild characters in any combination in a string. Wild characters may be redefined by using the DEFINE WILD CHARACTERS command. The LOCATE WILD command will locate the next "n" occurrences of the sting, and position the pointer to the last occurrence. If no "n" value is provided, the next occurrence will be located. If a "*" is substituted for n, all occurrences will be located, and the pointer will be positioned at the end of the file.

3.5.10 CHANGE WILD

CW [/stg1/stg2/ [n|n m]]

The CHANGE WILD command allows one to change a string using wild characters that will match other characters. Two wild characters are available, a "?" and a "*". A "?" in the string will match any other single character. A "*" in the string will match any number of characters (including zero). For example:

```
CW /SUBROUTINE * */*/ 100    will change

SUBROUTINE INPUT (IUNIT,CLINE,LTIME)
SUBROUTINE COMPUT (XARY,PIE,ILOCS,YARY)
SUBROUTINE ENDPRG

to

INPUT
COMPUT
ENDPRG
```

The first string may have up to 40 wild characters in any combination. The second string must have the same number, fewer, or no wild characters. If the same number or fewer wild characters are used in the second string, they must be in the same order as in the first string. For example,

```
CW /SUBROUTINE *(?ARY/SUBROUTINE *(XARY/      is legal
CW /SUBROUTINE *(?ARY/SUBROUTINE ?(*ARY/      is illegal
```

The optional m and n parameters correspond to the m-th occurrence of the string for the next n lines. The default values for m and n is 1. A "*" used for n or m denotes all lines from the current line to the bottom of file or all occurrences of "stg1", respectively. The line pointer will advance n-1 lines toward the bottom of the file.

3.5.11 DEFINE WILD CHARACTERS

DW [cs [cm]]

The DEFINE WILD CHARACTERS command allows one to change the wild characters used in the LOCATE WILD and CHANGE WILD commands. If no parameters follow the command, the current wild characters are displayed. The parameter "cs" must be a single character to use as the single wild character. The parameter "cm" must be a single character to use as the multiple character wild character. To specify "cm", "cs" must also be provided. The default single character wild character is "?" and the default multiple character wild character is "*".

3.5.12 CUT

CU c|/string/ [n]

The CUT command will divide one line into two lines after column c or after "string" for the next n lines. The cut may be either by column or by string location. After a cut, the current location is on the second half of the cut line. For example, "CU 80 *" will cut all lines into two lines after column 80 from the current line to the bottom of the file.

3.5.13 SPLICE

SP c|/string/ [n]

The SPLICE command causes the next line to be added on to the current line after either the column or string location given. This will be performed on n lines (default of 1 line) following the current line. For example, "SP 40" will add the next line on to the end of the current line after column 40.

3.5.14 TRUNCATE

TR c [n]

The TRUNCATE command truncates n lines to c columns, beginning at the current line. If n is omitted, only the current line is truncated. If a "*" is used for n, all lines from (and including) the current line will be truncated.

3.5.15 FREE

FR [OFF|VERIFY]

The FREE command causes input lines to be treated as free-field. The input lines that are affected include single line input, input mode (the prompt is "F>") and replace lines. If the VERIFY option is specified, the input lines are echoed to the terminal after formatting. A free-field input line consists of data elements, blanks, delimiters and strings. The fixed output fields are defined by the tab settings. Entering a FREE command changes the current tab settings to those most useful for generating input for several HEC programs, i.e.

2 8 16 24 32 40 48 56 64 72 80

1) A data field is a contiguous set of characters not containing a delimiter.

2) A delimiter is:
   a) One or more blanks
   b) , > <
   c) Single or double quotes for string delimiters ' "
   d) The tab character (default >)

3) A null data field is defined by two adjacent non-blank delimiters (ignoring blanks) and causes the corresponding output field to be blanked.

4) A string is a data field which begins with a string delimiter and contains all subsequent characters (including blanks and other delimiters) until the next occurrence of that delimiter. The beginning and ending string delimiters are not included in the string.

5) Input data fields are mapped into output fields. A data element may not be longer than its corresponding output field. A string may overflow its corresponding field and therefore span several fields.

6) Data fields are right justified into their corresponding output fields except when the ending delimiter (ignoring blanks) is < in which case the data field is left justified.

3.5.16 X and Y

```
X [/command/[command/[...]]|n]
Y [/command/[command/[...]]|n]
```

and
X1 through X5
Y1 through Y5

The X and Y commands allow the execution of several commands contained in one command. To define an X or Y "command string", enter X (or Y) followed by a space, then individual commands, each separated by a delimiter (e.g. a period "."). An unlimited number of commands may be placed in the X or Y command string. Once the command string has been defined, entering X (or Y) followed optionally by a count n will execute the defined command string n times. If n is not specified, it will default to 1.

X1 to X5, and Y1 to Y5 expand the number of X and Y commands that can be defined. The definition of the latest use of the X or Y commands will be retained until changed. The current definition will be displayed by preceding the command with an - sign, as -X or -Y2 .

3.6 Edit Control Commands

3.6.0 Summary

Name	Use	
HElp	HE	[command|number]
File Name	FN	[filename]
Full Screen	FS	[L][C]
Line Numbers	LN	[OFF]
Print Line number	PL	
Help Program	HP	[program-name|?|OFF|ON]
Help Variable	HV	[variable-name|line-id.field]
JUstify	JU	[DATA|OFF]
Job Control	JC	
Special Characters	SC	ON|OFF
TAb Settings	TA	[c][,c[...]]
Tab Character	TC	char
TErm definition	TE	[S=n] [L=m] [C=char]
Verify	V	[OFF]
Zone	Z	[m-n]

3.6.1 HELP

HE [command|number]

The HELP command provides on-line access to the COED documentation. When used without a parameter a help selection menu is displayed. When used with a 1 or 2 letter command identifier, information on that command will be displayed. When used with a number, various messages from the selection menu will be displayed.

3.6.2 FILE NAME

FN [filename]

The FILE NAME command allows the file name to be reset while editing a file. The new file name is used when a SAVE or FILE command is subsequently issued. If the file name is not specified, the current file name is displayed.

3.6.3 FULL SCREEN

FS [L] [C]

The FULL SCREEN command causes COED to operate in the full screen edit mode for certain terminals/computers. In this mode editing may be performed using the cursor keys to move around file. See the section on full screen editing for details.

On Harris computers, with a terminal that meets ANSI standards, the "L" parameter will cause the current line number to be displayed on the status line. The "C" parameter will cause the current column number to be displayed on the status line. These parameters can cause slower cursor movement on some terminals. Either (or both) parameters can be set in the COED terminal definition file (SYST*COEDTD).

3.6.4 LINE NUMBER

LN [OFF]

The LINE NUMBER command prints the line number along with each line. LN OFF turns off the line number printout.

3.6.5 PRINT LINE NUMBER

PL

The PRINT LINE NUMBER command prints the current line number.

3.6.6 HELP PROGRAM

HP [program-name|?|OFF|ON]

The HELP PROGRAM feature assists users in the preparation of data input files for HEC programs when using the full screen mode. This capability includes:

1) A prompt line at the bottom of the screen, indicating the names of variables for the current line (identified by the first two characters of the line).

2) Automatically setting tabs for the current line.

3) Automatically setting the justification for the current line.

4) Providing definitions of the current variable (where the cursor is) by pressing the HELP VARIABLE key (or the COMMAND key, then enter HV).

A general description of the HELP PROGRAM capabilities is included in the appendix.

The HP command followed by a question mark (?), or nothing, will display a list of those programs for which help is available. The HP, command followed by the name of one of the programs displayed, will activate help for that program. After help has been activated, it can be disabled by entering the command "HP OFF", and re-enabled with "HP ON".

3.6.7 HELP VARIABLE

HV [variable-name|line-id.field]

The HELP VARIABLE command provides definitions of HEC program variables as an aid in preparing data input files. HELP PROGRAM must be on in order to access this command.

In the full screen mode, HELP VARIABLE is requested by placing the cursor on the line and data field for which the variable help is desired, then pressing the HELP VARIABLE key. If no HELP VARIABLE key is available for the terminal you are on, press the COMMAND key, then type HV (return) to access help.

HELP VARIABLE may be used from the line edit mode by entering the command (HV), followed by either the variable name (e.g. HV METRIC), or the data line identifier and field number (e.g. HV J1.02).

3.6.8 JUSTIFY

JU [DATA|OFF]

The justify command controls how information is entered into fields in the full screen mode. When JUSTIFY is active (the default is off), and the cursor is on the right edge of a data field (by pressing the tab key), new characters entered cause those characters already in that field to be shifted to the left, so that entries are always aligned with the right side of the field. For more information see the Help Program Appendix.

When HELP PROGRAM is on, JUSTIFY is controlled by information in the HP program file, and cannot be turned off or on. JUSTIFY is normally used for entering data for programs that do not have a HP file.

3.6.9 JOB CONTROL

JC
DOS (on the PC)

The JOB CONTROL command temporarily moves you into Job Control, retaining your edit session. In Job Control, you can do limited things, such as list a file, get a directory (or map on the Harris). DO NOT edit any other files in this mode; If you do, your entire edit session may be lost.

On Harris computers, do not attempt a "FREE ALL". To return to your COED session, type "EXIT". You will be returned to the where you were when you issued the JC command.

On the PC, you are moved into an abbreviated version of DOS. In this version, there is no prompt (the cursor will remain at the end of the last line printed). To return to COED, press only a carriage return. The command "DOS" can be used instead of "JC" on the PC.

3.6.10 SPECIAL CHARACTERS

SC ON|OFF

The SPECIAL CHARACTERS command allows control characters to be entered, and will print control characters when encountered. Control characters are displayed as a caret (^) with the control character under it. For example, a control-G in a line would be printed as:

```
This is^ a test line.
       G
```

The COLUMN SCALE (CL) command will always print control characters, regardless if SPECIAL CHARACTERS is on or off. (Note: The delete key is not a control character, and its printing is dependent upon the terminal being used.)

3.6.11 TAB SETTINGS

TA [c1][,c[...]]

The TAB SETTINGS command sets tabs at columns c1, c2, etc. Up to twenty tab fields may be set. Tabs may be removed by typing "TA" with no parameters. Default settings are 7, 10, 13, 16, 19, 22 and 73. The tab settings are automatically changed to data style by entering a FREE command. The current tab settings may be displayed with the TERMINAL DEFINITION command.

On Harris computers, the default settings may be changed in the COEDTD file (see Site Installation).

3.6.12 TAB CHARACTER

TC char

The TAB CHARACTER command defines the character "char" to be used for tabbing. The default tab character is ">".

3.6.13 TERMINAL DEFINITION

TE [S=n] [L=m] [C=char]

The TERMINAL DEFINITION command describes the attributes of the individual terminal to the editor. This is separate from the terminal definition file used in full screen editing.

The "S=" option describes the line size to be used for printing. The default value is either 80 or 132 characters, depending on the file being edited.

The "L=" option specifies the number of lines per text line to print. The default value is 1 line.

The "C=" option defines the continuation character for input. This is useful when input lines are greater than the terminal column width. The default character is ">".

If TE is entered with no parameters, the current values of S, L, C and the current tab settings are displayed.

3.6.14 VERIFY

V [OFF]

The VERIFY command, when used in conjunction with the "OFF" parameter, suppresses output (unless explicitly requested using the PRINT command). The verification may be reinstated by entering V without "OFF".

3.6.15 ZONE

Z [m-n]

The ZONE command causes subsequent execution of the CHANGE and LOCATE commands to apply only to the zone (columns) specified. m is the starting column and n is the ending column. If no parameters are specified, the current zone settings are displayed.

4.0 Machine Specific Attributes

4.1 IBM PC (MS-DOS) Specifics

4.1.1 General Information

The P.C. version of COED requires MS-DOS Version 2.1 or later, and at least 512 KB of memory. The 8087 math co-processor is not needed. If COED will not run, the number of files specified in the "\CONFIG.SYS" file should be changed to 15, and the number of buffers set to 10, that is:

```
FILES=15
BUFFERS=10
```

The PC version has been tailored to take advantage of the speed of in-core editing. If the file being edited exceeds available memory, a spill file will be used (this occurs at about 2000 lines at 80 columns). When the spill file is needed, editing speed performance is degraded, so appropriately sized files is recommended.

4.1.2 Files Used

When a file is edited, two backup files are created for recovery purposes. By default, these files will be generated in the root directory, unless an accessory file directory has be defined as described in section 4.1.3. Their names are:

```
COED.BUP
COED.REC
```

The file COED.BUP will contain a copy of the last file edited, before any modifications were made. This file can be used, if unwanted changes are accidently saved.

The file COED.HLP must be available for HELP features to function. The file \COED.TRM will be created in the root directory of the default drive if any setup features are used (see below).

4.1.3 Setup Function for Full Screen

The control-F2 function key in the full-screen edit mode provides a means of setting screen colors (for color monitors), defining the directory for the COED accessory files, and making COED come up in the full screen mode. After pressing control-F2, a short menu will appear, requesting the user to enter a "P", or a "N" to set the Full Screen mode to Permanent (or unset it), a "D" to change the accessory file directory, or a "C" to change screen colors. If a "P" is entered, then COED will always go directly in to the Full Screen mode at the beginning of an edit session, without the user having to type "FS". To reset this feature, so that COED will begin with the line edit mode, type a "N". The user is

returned to the full screen mode when either of these characters are pressed.

If the accessory file directory is changed, then COED will generate the recovery files in this directory, and look for the help file (COED.HLP) here. After setting the directory, and terminating the edit session, the user should copy all of these files from the old directory (the default is the root directory) into the new directory, with the exception of the file \COED.TRM. COED.TRM must always remain in the root directory.

To change the screen colors, enter a "C" from this menu. This will cause another short menu to appear, allowing one to change the foreground and background colors for the main screen, the status line, and the help program prompt line on a color screen. These colors are retained between edit sessions (in file \COED.TRM). By pressing the "F" key, the colors scroll through the foreground color (16 colors). "B" changes the background color (8 colors), "S" changes the status line foreground color, and "T" changes the status line background color. A "H" toggles through the help program prompt line foreground colors, while a "P" toggles through the prompt line background colors. When complete, a carriage return returns to the edit mode.

4.2 Harris Specific Information

If $ADD is set to "ON", COED will turn if off and print a message to that effect. To edit a file with $ADD ON, the "A" option must be used ($ADD does not have to be set on, COED will automatically set it to on). For example:

```
COED.A  MYFILE
```

COED uses the file "SYST*COEDTD", which must be located in the system qualifier, to identify parameters (such as the duplex), and and terminal definitions for full screen editing. Located in this file is the qualifier name of where additional accessory files may be found (such as COEDHE, for on-line documentation, and the help program files). See the Harris installation section for more information on the COEDTD file.

In order to achieve the most effective performance on Harris mini-computers in use by the Corps of Engineers, several COED capabilities have been tailored for that system. Most editing is done in-core with the operating system providing virtual memory paging through the in-core 'file' instead of using less efficient user disk I/O. This makes COED require a larger amount of memory than previously. Some system site managers are reluctant to provide users with permission to use more memory under the belief that it will have detrimental effects on their system performance. Our testing has shown that limiting the program size may cause less efficient use of computer resources. The additional overhead of the user disk I/O implementation is far more demanding on system resources than the currently implemented virtual memory approach.

APPENDICES

Appendix A
Alphabetical Command Summary

Command	Use	
Alter	AL	
Bottom	B	
Change	C	[/stg1/stg2/ [n|n m]]
Change Wild	CW	/stg1/stg2/ [n|n m]
Column Duplicate	CD	m-n c [r]
Column Move	CM	m-n c [r]
Column Remove	CR	m-n [r]
CoLumn scale	CL	
Column Set	CS	m[-n] 'char' [r]
CUt line	CU	c|/string/ [n]
DElete	DE	[n]
Define Wild	DW	[cs] [cm]
DUplicate	DU	
ENd	EN	
Find	F	[string]
Find Except	FE	string
FILE	FILE	[filename]
File Name	FN	[filename]
FRee	FR	[OFF|VERIFY]
Full Screen	FS	[L][C]
GEt	GE	[filename] [m|m n]
Goto	G	n
HElp	HE	[command|number]
Help Program	HP	[program-name|?|OFF|ON]
Help Variable	HV	[variable-name|line-id.field]
Insert	I	[line]
Job Control	JC	
JUstify	JU	[DATA|OFF]
Line Numbers	LN	[OFF]
Locate	L	[/string/ [n]]
Locate Except	LE	/string/ [n]
Locate Wild	LW	/string/ [n]
MOve	MO	
Next	N	[n]
Overlay	O	line
Print	P	[n|ALL]
Print Line number	PL	
PUt	PU	[filename]
QUit	QU	
Replace	R	line
REmove	RE	
STart	ST	

SAVE	SAVE	[filename]
Special Characters	SC	ON|OFF
SPlice lines	SP	c|/string/ [n]
TAb settings	TA	[c][,c[...]]
Tab Character	TC	char
TErm definition	TE	[S=n] [L=m] [C=char]
Top	T	
TRuncate	TR	c [n]
Up	U	[n|/string/]
Verify	V	[OFF]
X	X	[/command/[command/[...]]|n]
Y	Y	[/command/[command/[...]]|n]
Zone	Z	[m-n]

Appendix B
Table of COED Commands

AL	-ALter	FR	-FRee	PL	-Print Line number
B	-Bottom	FS	-Full Screen	PU	-PUt
C	-Change	G	-Goto	QU	-QUit
CD	-Column Duplicate	GE	-GEt	R	-Replace
CL	-CoLumn scale	HE	-HElp	RE	-REmove
CM	-Column Move	HP	-Help Program	SAVE	-SAVE
CR	-Column Remove	HV	-Help Variable	SC	-Special Characters
CS	-Column Set	I	-Insert	SP	-SPlice lines
CU	-CUt line	JC	-Job Control	ST	-STart
CW	-Change Wild	JU	-Justify	T	-Top
DE	-DElete	L	-Locate	TA	-TAb settings
DU	-DUplicate	LE	-Locate Except	TC	-Tab Character
DW	-Define Wild	LN	-Line Numbers	TE	-TErm definition
EN	-ENd	LW	-Locate Wild	TR	-TRuncate
F	-Find	MO	-MOve	U	-Up
FE	-Find Except	N	-Next	V	-Verify
FILE	-FILE	O	-Overlay	X	-X
FN	-File Name	P	-Print	Z	-Zone

Appendix C

Help Program Information

This section describes the COED Help Program capabilities that aid in generating and editing data input files for engineering programs. The following topics are covered in this section:

1) Prompting with data input line definitions.
2) Automatic tab stops for each data input line.
3) Automatic justification of data in each field.
4) Checking for non-numeric data at time of entry.
5) Use of the numeric key pad for data input.
6) Data input variable definitions.
7) Program Help files.

C.1 Data Input Line Prompts

The first two characters of the line that the cursor is on governs the contents of a program prompt line located at the bottom of the screen. These characters are compared against a list of line identifiers from a file containing that program's help information. If the identifier is recognized, the prompt line associated with that identifier is displayed. If the identifier is not recognized, a default prompt is displayed. The prompt is changed if the identifier on the current line is changed, or if the cursor is moved to a different line.

C.2 Automatic Tab Stops

The tab stops are dynamically reset for each line, according to the line identifier and information from the program's help file. If the line identifier is not recognized, then default tab stops are set.

C.3 Automatic Justification

With Help Program on, the justification setting is dynamically set for each line and field, according to the line identifier and information from the program's help file. When Help Program is off, the justification mode may be set by the JUSTIFY key or command. Without data justification on, data entry and tab stops operate normally. With data-justify on, characters entered are right justified within each field. If the cursor is on the right edge of a field, any characters typed are aligned to the right of the field. If the cursor in not on the right edge of the field, then any characters typed will be placed normally, until reaching the right edge. The tab key is used to move to the next field, whereby the cursor is placed on the right edge of the field. When data justification is on, the backspace and the delete key will remove the character at the cursor location.

C.4 Checking for Non-numeric Data

When entering data, the data may be checked to assure only numeric digits are entered. This can prevent the entry of the letter "O" when a zero was intended. When numeric data checking is perfo:med, only the following fourteen characters may be entered: "0123456789 .-+". Numeric data checking is governed by information in the program's help file.

C.5 Use of the Numeric Keypad

Numeric data may be entered most rapidly from the numeric keypad. Unfortunately, on many keyboards this disables the use of the keypad for cursor movement. The keypad may be set in a numeric mode by pressing the COED "NUM-LOCK" key. To return the keypad to a cursor movement operation, press the COED "NUM-LOCK" key again. On personal computers with programs that emulate terminals, the local keyboard "NUM-LOCK" key may need to be set on, and the separate COED "NUM-LOCK" key is used to toggle between the cursor movement application and numeric application.

C.6 Variable Definitions

If the Help Program mode is on, and the program's help file contains variable definitions, the definitions may be displayed on the screen by use of the HELP VARIABLE command.

In the full screen mode, HELP VARIABLE is requested by placing the cursor on the line and data field for which the variable help is desired, then pressing the HELP VARIABLE key. If no HELP VARIABLE key is available for the terminal you are on, press the COMMAND key, then type HV (return) to access help.

HELP VARIABLE may be used from the line edit mode by entering the command (HV), followed by either the variable name (e.g. HV METRIC), or the data line identifier and field number (e.g. HV J1.02).

C.7 Help Program Files

The Help Program feature uses an external file, for each program, that contains information on the line variable names, tab settings, justification, non-numeric checking, and variable definitions. The file "COEDHP" contains a list of all programs for which help is available, and the names of each program's help file. (The format of the COEDHP file is self-explanatory.) The COEDHP file is located in the directory or qualifier defined in the 0000SYST*COEDTD or \COED.TRM file.

The program help files are indexed text files. If a progran help file is changed, the byte or line count that is used for indexing may become incorrect, and the help for that program will not operate properly. If you wish to modify or add a help file, contact the HEC for assistance.

Appendix D

Harris Installation Information

Several default settings may be specified in the file "SYST*COEDTD", along with terminal definitions for full-screen editing. These settings include: 1) the duplex (echoplex); 2) the column size; 3) tab settings; 4) the qualifier location for the help file; and 5) the terminal definition to be used for full-screen editing.

D.1 Setting General Editing Defaults.

In order to change the default settings, the first four characters of the first line in file "SYST*COEDTD" must be LOAD. On subsequent lines, instructions can be given to change the default settings. This typically consists of a four character keyword, followed by an equal sign, then the new setting. Comments may be inserted (as separate lines) by using an asterisk (*) as the first character in the line. The settings must end with an ENDLOAD statement.

D.1.1 Duplex (Echoplex)

By default, COED checks the Harris string register "SDX" to determine if the duplex is FULL or HALF (by checking a register, different terminals may have different duplexes). If that register is nonexistent, full duplex is assumed. The COEDTD file may specify a different register to check, or "hardwire" the duplex. If the duplex is half, and full-screen editing is desired, see the notes under Full-Screen Terminal Definition that follows.

To have COED check another register, use the keyword "DREG", followed by an equal sign (=), then the register name. For example, to check the register "DPX", use:

```
DREG=DPX
```

To "hardwire" a duplex setting, use the keyword "DUPL", followed by an equal sign (=), then the word "FULL" or "HALF". For example:

```
DUPL=HALF
```

D.1.2 Column Size (S=value)

The default column size (length) is set to the optimum 80 characters. With this setting, COED will check the first 100 lines, to see if any lines are longer than 80 characters. If so, the column size will be set to 132 characters. Any value specified on the execution line will override this. If a default column size other than 80 is desired, it may be specified by the keyword "S", followed by an equal sign, then the new value. For example:

```
S=132
```

[In this case, COED will not check the first 100 lines.] In any case, if the user specifies an S value on the execution line, it will override any setting in the COEDTD file.

D.1.3 Tab settings

The default tab settings (7, 10, 13, 16, 19, 22, and 73) may be re-specified by giving a tab command in this section. This is done by entering the letters "TAB" followed by the new tabs. For example:

```
TAB 7, 11, 18, 30, 73
```

D.1.4 Qualifier Containing Help File

A separate qualifier may be specified for the location of the COED help file (COEDHE), and the help program files (COEDHP). This is accomplished by giving the letters "QUAL", followed by an equal sign, then the qualifier. For example:

```
QUAL=1000COED
```

If no qualifier is given, the default of "SYST" is used.

D.1.5 Terminal name for Full Screen

When a user first goes into full-screen mode, COED will check the register "STM" to identify the type of terminal being used. If that register is not present, a list of the terminals defined will be given, and the user will asked for the name of the terminal in use.

A different register can be checked, or the terminal name may be hardwired (useful only where a site has one type of terminal). To have COED check a different register, use the keyword "TREG", followed by an equal sign (=), then the 3 character register name. The register should contain a four character name or

abbreviation for the terminal. For example to have COED check the register "TRM":

```
TREG=TRM
```

To "hardwire" COED so that only one type of terminal can be used, use the keyword "TERM", followed by an equal sign (=), then the 4 character terminal name or abbreviation. For example, to set the terminal name to "TABG":

```
TERM=TABG
```

D.2 Example Top Portion of the "SYST*COEDTD" File

```
LOAD
* SET THE DUPLEX REGISTER TO "DUP"
DREG=DUP
* SET THE ACCESSORY FILE QUALIFIER TO HLIB
QUAL=HLIB
* RESET THE DEFAULT COLUMN SIZE (LENGTH) TO 132:
S=132
* RESET THE DEFAULT TABS:
TAB 7 11 20
* SET THE TERMINAL REGISTER TO "TRM":
TREG=TRM
ENDLOAD
```

D.3 Terminal Definitions for Full-Screen Editing

In order to use full-screen features on the Harris, the terminal type being used must be defined in the file "SYST*COEDTD". The definitions include the number of lines and columns the terminal screen has, any messages that need to be sent to the terminal to set it to the proper mode, or reset it when done, down-loading information to any programmable softkeys, redefining any keys, and providing any help messages to the user.

Currently, full-screen editing can only be done on terminals that meet ANSI terminal standards (this includes most newer terminals). Unfortunately, many Harris terminals, such as the Beehive, do not meet these standards. It is planned that a future release will have provisions for those terminals.

Full-screen editing can only work in a full duplex mode. If you use half duplex, a message can be sent to some terminals to change them into full duplex during full-screen editing, then back to half duplex during the regular line edit mode. For more information, see the "BM" and "EM" keywords (following).

A terminal definition begins with the keyword "TERM" (starting in the left-most column), followed by a blank then the terminal name (only the first four characters are used). If more than one model of terminal meets the same definition, one or more TERM statements can follow this. A terminal definition ends with the keyword "ENDTERM" (starting in the left most column). For example:

```
* Define Tektronix 4105, 4017, and 4109
TERM 4105
TERM 4107
TERM 4109
(definitions)
ENDTERM
```

Each terminal definition must contain at least the following keywords: "TERM", "ENDTERM", "NL", and "NC". Except for TERM and ENDTERM, each keyword consists of two upper case characters. All parameters are separated from the keyword by a blank.

D.3.1 Keyword Definitions.

a) TERM: Identifies the beginning of a terminal definition. The terminal name must follow, of which only the first four character are used.

b) ENDTERM: Defines the end of a terminal definition.

c) NL: Number of lines the terminal screen has. This number may be followed by the word "ON", which indicates the current file line number should be displayed on the status line. On most terminals at most baud rates it takes too long to change the line number while moving through the file. Example:

```
NL 24 ON
```

d) NC: Number of column the terminal screen has. If the terminal has two mode (e.g. 80 columns or 132 columns) or if the terminal can scroll horizontally, use the larger number (e.g. 132). This number may be followed by the word "ON", which indicates the column number should be displayed on the status line. On most terminals at most baud rates it takes too long to change the column number. Example:

```
NC 132
```

e) IM: Initiate Message. Upon entering full-screen mode (for the first time), the message that follows will be sent to the terminal. Most of the time this message is to set the terminal in a mode necessary to do full-screen editing. The message is defined by a beginning and ending delimiter (e.g. a quote ('), or dollar sign ($)), which may not appear anywhere else in the message. Control characters are sent by using a caret (^) preceding the character. For example, a ^A will send a control-A, a ^[will send an escape character. To send a caret, supply two carets (e.g. ^^). The message is sent exactly as is; No carriage return or line feed is appended. For example:

```
* Set terminal to ANSI mode
IM '^[<'
```

This will send an escape character followed by a less-than sign character (<). Also:

```
* Set horizontal scroll on
IM '^[[=4h'
```

This will send an escape character followed by a left bracket ([) followed by an equal sign (=), a 4, and a lower case h.

The initiate message is often used to send information to the terminal for programmable soft keys. There may be as many initiate messages as needed.

f) RM: Reset message. The message following the keyword is sent upon termination of COED. The message follows the same conventions used above. This is often used to return the terminal to its initial state.

g) BM: Begin Message (this keyword is used infrequently). The Begin Message is typically used for setting the duplex to full upon entry of full-screen on a half duplex system. The message following the keyword will be sent to the terminal every time COED goes from a line-edit mode to a full-screen mode. The same conventions used in "IM" are used here. Only 5 "BM's" may be used per terminal. [An "IM" is sent only the first time COED enters full-screen mode during an edit session. A "BM" is sent every time COED enters full-screen mode.]

h) EM: End Message. Used in accordance with Begin Message. The message following the keyword is sent to the terminal every time COED goes into the line edit mode from full-screen model.

i) HE: Help. The message following the HE keyword is sent to the terminal when the user requests help from full-screen mode. Typically a series of HE's are provided to fill up the screen to indicate to the user what keys or control characters do what functions. The messages follow the conventions given under "IM", except that a carriage return and a line feed are appended to each message.

j) HK: Help Key. This provides the name of the help key, which is displayed on the status line. Up to four characters may be used as the help key name. Example:

```
HK F22
```

D.4 Redefining Keys for Full-Screen Functions

Many terminals provide alternative keys to provide functions such as cursor movement, or the function keys may not be programmable. By redefining the COED full-screen function sequence these keys can be utilized. The keys to be used must either send out a single control character, or an a control code sequence of up to 5 characters (a control character followed by up to 4 more characters). All sequences for a particular control character must be the same length.

The keywords for redefining the function sequence are DEFINE and DEFINE2. DEFINE will add on that key's definition, and remove any other definitions for that function, while DEFINE2 will add on to that key's definition. There may be several definitions for each function (there is room for a total of 200 definitions).

Suppose that the terminal you are installing has 4 keys, separate from the key pad, with arrows on them; one right, one left, one up, one down. If you want the key with the left arrow to preform a cursor left, and that key produces an "escape [D" when pressed, the COED cursor left function can be defined by providing the following line in the definition file:

```
* Set the alternative cursor-left key
DEFINE2 CURL=^[[D
```

The other cursor left key will remain defined (by default) as the 4 on the key pad. If DEFINE only were used, the keypad 4 would preform no function.

If you wish the backspace to move the cursor left (a non-destructive backspace) without blanking the character, you can add another definition to the cursor left:

```
* Make the Backspace Non-Destructive
DEFINE2 CURL=^H
```

(The backspace key sends a control-H when pressed). For these examples, a total of 3 keys performed the cursor left function.

Several functions can be combined into one key in order to make more powerful keys. For example, to create a key to clear to the end of the line, you could program a softkey (terminal dependent) to send the delete function, a carriage return, then another delete key. However, COED limits its type-ahead capability only to standard characters (if someone were to keep the "Page Down" key pressed, it would continue paging down long after the key was released if type-ahead was used). Type-ahead may be turned completely on by sending an escape then ++, and back off by sending a escape then --. In order to make a multiple function key, type-ahead must be first turned on (and reset to off when done). For example to download a "clear to end of line" sequence to a terminal softkey, the following Initiate Message may be sent:

```
* Download Clear to end of line key (multiple function key)
* Sequence: Type-ahead on, delete char, carriage return,
* delete char, type-ahead off.
IM '^[++^[On^M^[On^[--'
```

D.5 Define Function Keywords.

The keywords (and the default character sequences that invoke that function) to used when redefining keys for functions (section D.4) are provided in the following table. In this table a caret (^) indicates that the next letter is a control character. For example a ^A represents control A and a ^[represents the escape character.

Keyword	Default Sequence	Function
COMN	^A	Command (single)
LINE	^B	Line-Edit
CURR	^[Ov	Cursor Right
CURL	^[Ot	Cursor Left
CURU	^[Ox	Cursor Up
CURD	^[Or	Cursor Down
PGUP	^[Oy	Page Up
PGDN	^[Os	Page Down
HOME	^[Ow	Home
END	^[Oq	End
BOL	see below	Beginning of Line
EOL	see below	End of Line
TABR	^I	Tab Right
TABL	none	Tab Left
INSC	^[Op	Insert Character Mode
DELC	^[On	Delete Character
INSL	^F	Insert Line Mode
DELL	^G	Delete Line
HELP	^D	Help
REST	^C	Restore
SCUP	^[Om	Scroll Up
SCDN	^[Ol	Scroll Down
NUML	^T	Numb-Lock
NXTW	^[OM	Next Word
PREW	see below	Previous Word
TOS	see below	Top of Screen
BOS	see below	Bottom of Screen
SAVE	none	Save File
FILE	none	File
QUIT	none	Quit
KEY5	^[Ou	Key 5 on the Keypad
JUST	none	Justify
HLPV	none	Help Variable
EFLD	none	Erase Field
REFR	none	Refresh Screen

Several functions (such as Beginning of Line) use 2 keys for their default definition: the keypad 5, then another keypad key. These may be refined to use only one key, if desired.

D.6 ASCII Control Character Code Representations

Decimal Value	Mnemonic	COED Representation
00	NUL	^@
01	SOH	^A
02	STX	^B
03	ETX	^C
04	EOT	^D
05	ENQ	^E
06	ACK	^F
07	BEL	^G
08	BS	^H
09	TAB	^I
10	LF	^J
11	VT	^K
12	FF	^L
13	CR	^M
14	SO	^N
15	SI	^O
16	DLE	^P
17	DC1 (X-ON)	^Q
18	DC2	^R
19	DC3 (X-OFF)	^S
20	DC4	^T
21	NAK	^U
22	SYN	^V
23	ETB	^W
24	CAN	^X
25	EM	^Y
26	SUB	^Z
27	ESC	^[
28	FS	^\
29	GS	^]
30	RS	^/
31	US	^_
127	Delete	^?

D.7 Example Terminal Definition.

The following provides an example terminal definition used for the TAB 132/15-G terminal.

```
TERM TABG
NL 24
NC 132
*  Define alternative cursor keys
DEFINE2 CURU=^[[A
DEFINE2 CURD=^[[B
DEFINE2 CURL=^[[D
DEFINE2 CURR=^[[C
DEFINE2 HLPV=^V
*
* Define backspace as non-destructive (i.e. cursor left)
* DEFINE2 CURL=^H
*
*  Set terminal to ANSI mode
IM '^[<'
*  Set "TABMODE" on
IM '^[_TABON^[\'
*  Set Horizonal scroll on
IM '^[[=4h'
*  Set Black Background
IM '^[[?5l'
*  Set normal attributes
IM '^[[0m'
*  Set replacement mode (instead of insert)
IM '^[[4l'
*  Set alternative Keypad on
IM '^[='
*
*  Set Soft key ledgends
IM '^[_L1Delete Ln^[\'
IM '^[_L2Insert Ln^[\'
IM '^[_L3 Num-Lock^[\'
*  Ledgened 4 is reserved for future use
IM '^[_L4         ^[\'
IM '^[_L5  Help   ^[\'
IM '^[_L6 Restore ^[\'
IM '^[_L7 Command ^[\'
IM '^[_L8Line-Edit^[\'
```

```
*  Set Soft Key Messages
IM '^[_M1^G^[\'
IM '^[_M2^F^[\'
IM '^[_M3^N^[\'
IM '^[_M4^[\'
IM '^[_M5^D^[\'
IM '^[_M6^C^[\'
IM '^[_M7^A^[\'
IM '^[_M8^B^[\'
*  Display these settings
IM '^[9'
*
*  TAB Reset Messages
RM '^[[=6h'
RM '^[>'
RM '^[_TABOFF^[\'
* TAB HELP MESSAGES
* TAB Help Key is S5
HK S5
HE '+-------------------------------------------------------------------------+'
HE '|   -------------- TAB TERMINAL COED FUNCTION KEYS ------------------   |'
HE '|   | S1 - Delete current line          S5 - HELP                   |   |'
HE '|   | S2 - Insert line mode (toggle)    S6 - Restore line           |   |'
HE '|   | S3 - Number-Lock                  S7 - Command                |   |'
HE '|   | S4 - (reserved)                   S8 - Line edit mode         |   |'
HE '|   ---------------------------------------------------------------   |'
HE '|                                      K E Y  P A D  K E Y S          |'
HE '|                                    ------------------------------   |'
HE '|     Other Keypad Functions         | HOME |  ^^  | PAGE | SCROLL |  |'
HE '|                                    |      |  |   |  UP  |   UP   |  |'
HE '|   Key 5 then 4:  Beginning of line |------------------------------|  |'
HE '|   Key 5 then 6:  End of line       | <--- |      | ---> | SCROLL |  |'
HE '|   Key 5 then 2:  Bottom of screen  |      |  5   |      | DOWN   |  |'
HE '|   Key 5 then 8:  Top of screen     |------------------------------|  |'
HE '|   Key 5 then "print":  Previous    |  END |  |   | PAGE |        |  |'
HE '|                        Word        |      |  V   | DOWN | NEXT   |  |'
HE '|                                    |-------------------- WORD   |  |'
HE '|                                    |    INSERT     | DEL  |        |  |'
HE '| Press <space> for more help        |     CHAR      | CHAR |        |  |'
HE '| Press <S5> (HELP) to resume edit   ------------------------------   |'
HE '+-------------------------------------------------------------------------+'
ENDTERM
```

Appendix E

MS-DOS P.C. Installation Information

The MS-DOS version of COED can either be used on a dual-floppy system, or on a hard disk system. On either system, at least 460 Kbytes of free memory (equivalent to 512 Kbytes of total memory running DOS 3.1) and DOS 2.1 (or later) are required. The math coprocessor is not needed. Make sure that the \CONFIG.SYS file has at least the following: FILES=15 and BUFFERS=10.

Three diskettes are supplied for COED. The first diskette contains the executable program, the on-line help file, some installation setup files, and some accessory files. The second diskette contains the HEC help program files, for use with the help program feature of COED. The third diskette contains a copy of the COED documentation, if a hard copy is not already available.

Included with the first diskette is a Crosstalk file (HARRIS.XTK) that emulates a Televideo terminal for communication with a Harris computer. Since the Televideo meets ANSI terminal standards, it can be used for full-screen editing. This file may be used by those, with Crosstalk, for full-screen editing by pressing the NUM LOCK key to enable use of the key pad (to communicate with the Crosstalk program, press the NUM LOCK key a second time). Communication parameters (e.g., baud rate, port) may need to be changed.

On a dual-floppy system, insert the first COED diskette into one of the drives, and edit file(s) on the other drive. The default drive may be either drive, as COED will check both drives for the COED.TRM file (the file that remembers the screen colors). (Make sure that no directory is set in the COED.TRM file).

The following are instructions for installing COED on a hard disk (assumed to be drive C: in these examples). The instructions in section one are for loading the COED program into the root directory, and the accessory files into directory C:\UTIL. Section two provides instructions on loading the COED accessory files into an alternative directory. If you have an old version of COED, backup that version, then erase all the old COED files on the hard disk before you begin.

1) A batch file (SETUP-C.BAT) has been included on diskette 1, which will load COED.EXE and COED.TRM into the hard disk root directory, then copy the accessory files into the directory \UTIL (which will be created, if not present). To install COED in this manner, insert diskette one into drive A, then type: A:SETUP-C and follow the instructions on the screen.

2) If you desire to load the COED accessory files (i.e. help files, recovery files, and help program files) in a alternative directory, complete the following instructions:

   a) Create the directory for the accessory files if it does not yet exists (e.g., type: MD \UTIL).

   b) Place the first COED diskette in the A drive and change the default drive to A: (type: A: in DOS).

   c) Use COED to edit an dummy file (e.g., type: COED XYZ).

   d) After the E> prompt has appeared, enter into full-screen mode (type: FS). Now press the setup key (control-F2).

   e) Set the drive and the alternative directory by pressing the "D" key. Enter the drive and directory selected (e.g., type: C:\UTIL\).

   f) Exit COED by pressing the quit key (control F10).

   g) Change the default drive back to the hard disk drive (type C:) then copy A:COED.TRM into the root directory of the hard disk (type: COPY A:COED.TRM C:\COED.TRM).

   h) Copy the COED.HLP file from diskette one into the directory selected (e.g., type: COPY A:COED.HLP C:\UTIL).

   i) Copy the COED program from diskette one into the root directory (or alternative directory according to the note below) (e.g., type: COPY A:\COED.EXE \COED.EXE).

   j) Copy all of the files on diskette two into the directory selected (e.g., type: COPY A:*.* C:\UTIL).

3) If you desire to store all the COED files in your root directory, copy all of the COED files from diskettes one and two into the root directory (e.g., type: COPY A:*.* C:\). The setup files may be erased after copying.

Notes:

The COED documentation file on diskette three does not need to be placed on the hard disk (only use it if you desire a hard copy of the documentation).

The COED executable may optionally be stored in an alternative directory by setting the path to that directory. This is done by modifying the autoexec.bat file to contain a PATH command that includes the directory where the executable is. For example, if COED is to be stored in "\UTIL", add (or modify) the following line to autoexec.bat:

```
PATH C:\;C:\UTIL
```

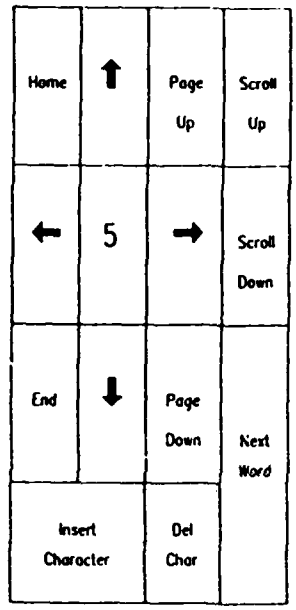
See your DOS Documentation for further information on the PATH command.

Copy the executable file (COED.EXE) from diskette one into this directory.

Appendix F
Full Screen Key Templates

MS-DOS Personal Computer

COED FULL SCREEN KEYS

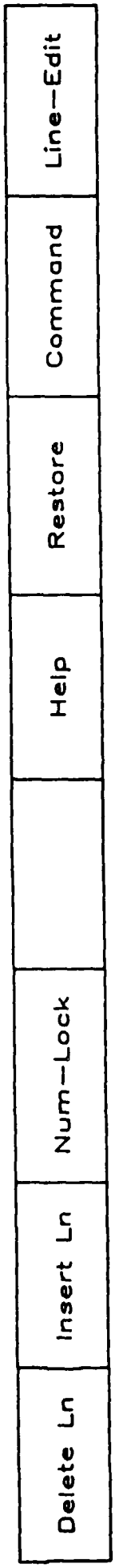


Beg of Line: 5 ←
End of Line: 5 →
Top of Screen: 5 ↑
Bot of Screen: 5 ↓
Prev Word: 5 Nxt Wd

COED FUNCTION KEYS

Key Control + Key Shift + Key Alt + Key	
Help – – Help Var.	Restore PC Setup – –
Delete Line – Erase Field –	Insert Line – – –
– – – –	– – – –
– – – –	– – – –
Command – – –	Line-Edit Quit Save File

TAB Terminals



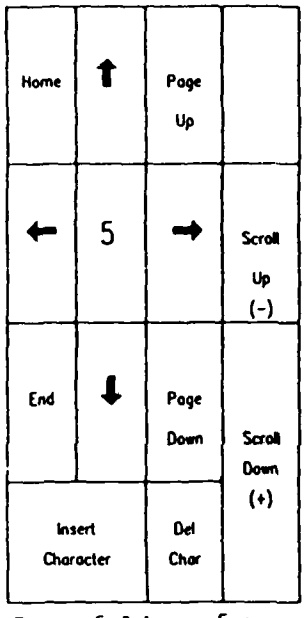
COED FULL SCREEN KEYS

Home	↑	Page Up	
←	5	→	Scroll Up (-)
End	↓	Page Down	Scroll Down (+)
Insert Character		Del Char	

Beg of Line: 5 ←
End of Line: 5 →
Top of Screen: 5 ↑
Bot of Screen: 5 ↓

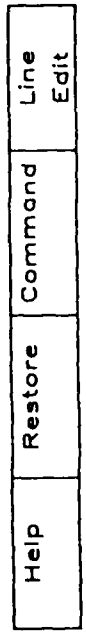
Next Word: Cntl →
Prev Word: Cntl ←

COED FULL SCREEN KEYS

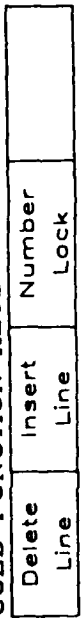


Beg of Line: 5 ←
End of Line: 5 →
Top of Screen: 5 ↑
Bot of Screen: 5 ↓

Next Word: Cntl →
Prev Word: Cntl ←



COED FUNCTION KEYS



Human Design Systems (HDS) Terminals

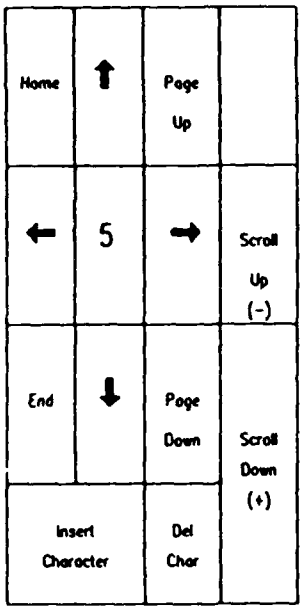
COED FUNCTION KEYS F1-F5

		Delete Ln	Refresh	Save	Shift
Insert Ch	Delete Ch	Insert Ln	Del to EOL	File	

COED FUNCTION KEYS F17-F23

			Help Var			Shift
Delete Ln	Insert Ln	Num-Lock	Help	Restore	Command	Line Edit

COED FULL SCREEN KEYS



Beg of Line: 5 ←
End of Line: 5 →
Top of Screen: 5 ↑
Bot of Screen: 5 ↓

Next Word: Cntl →
Prev Word: Cntl ←

INDEX